

1
2
3 **UNITED STATES DISTRICT COURT**
4 **NORTHERN DISTRICT OF CALIFORNIA**
5 **SAN JOSE DIVISION**
6

7 FINJAN, INC.,
8 Plaintiff,
9 v.
10 CISCO SYSTEMS, INC.,
11 Defendant.
12

Case No. 17-cv-00072-BLF

**ORDER CONSTRUING CLAIMS IN
U.S. PATENT NOS. 6,154,844; 6,804,780;
7,647,633; 8,141,154; 8,677,494**

[Re: ECF 100, 112, 127]

13
14 Plaintiff Finjan, Inc. (“Finjan”) brings this patent infringement lawsuit against Defendant
15 Cisco Systems, Inc. (“Cisco”), alleging infringement of five of Finjan’s patents directed to
16 computer and network security: U.S. Patent Nos. 6,154,844 (“the ’844 patent”); 6,804,780 (“the
17 ’780 patent”); 7,647,633 (“the ’633 patent”); 8,141,154 (“the ’154 patent”); and 8,677,494 (“the
18 ’494 patent”) (collectively, the “Asserted Patents”). The Court held a tutorial on June 7, 2018 and
19 a *Markman* hearing on June 15, 2018 for the purpose of construing ten disputed terms in the ’844,
20 ’780, ’633, ’154, and ’494 patents.

21 **I. BACKGROUND**

22 The Asserted Patents are directed to network security technologies that detect online
23 threats from malware. Finjan asserts that Cisco’s products and services infringe the Asserted
24 Patents. *See generally* Second. Am. Compl., ECF 55. Each patent is summarized below.

25 **A. The ’844 Patent**

26 The ’844 patent is titled “System and Method for Attaching a Downloadable Security
27 Profile to a Downloadable” and was issued on November 28, 2000. Ex. 1 to Hannah Decl. (the
28 ’844 patent), ECF 100-3. This patent claims systems and methods for inspecting Downloadables

for suspicious code or behavior according to a set of rules and generating a profile of the results from the inspection. *See, e.g., id.* at 1:62–3:7. In some embodiments, a content inspection engine generates a security profile and links that profile to a Downloadable. *Id.* at 2:3–11. The profile can include certificates that are later read by a protection engine to determine whether or not to trust the profile. *Id.* at 2:20–48. By providing verifiable profiles, the claimed systems and methods may efficiently protect computers from hostile Downloadables. *Id.* at 2:61–3:7.

B. The '494 Patent

The '494 patent is titled “Malicious Mobile Code Runtime Monitoring System and Methods” and was issued on March 18, 2014. Ex. 2 to Hannah Decl. (the '494 patent), ECF 100-4. The patent provides “[p]rotection systems and methods . . . for protecting one or more personal computers (“PCs”) and/or other intermittently or persistently network accessible devices or processes from undesirable or otherwise malicious operations” *Id.* at 2:51–55. To achieve this goal, some embodiments utilize a protection engine in order to identify executable code. *Id.* at 11:65–12:14, 12:38–47.

C. The '780 Patent

The '780 patent is titled “System and Method for Protecting a Computer and a Network From Hostile Downloadables” and was issued on October 12, 2004. Ex. 3 to Hannah Decl. (the '780 patent), ECF 100-5. This patent teaches the generation of a re-usable ID for downloaded files so that future iterations of those files can be easily identified. For instance, the patent discloses that an ID generator can compute an ID that identifies a Downloadable by fetching components of the Downloadable and performing a hashing function on the fetched components. *See, e.g., id.* at 2:12–16.

D. The '633 Patent

The '633 patent is titled “Malicious Mobile Code Runtime Monitoring System and Methods” and was issued on January 12, 2010. Ex. 4 to Hannah Decl. (the '633 patent), ECF 100-6. The patent provides systems and methods for protecting devices on an internal network from code, applications, and/or information downloaded from the Internet that performs malicious operations. *Id.* at Abstract. At a high level, some embodiments include a protection engine that

resides on a network server and monitors incoming information for executable code. *Id.* at 2:20–3:4. Upon detection of executable code, the protection engine deploys a “mobile protection code” and protection policies to a downloadable-destination. *Id.* col. 3:5–21. At the destination, the Downloadable is executed, typically within a sandboxed environment, and malicious or potentially malicious operations that run or attempt to run are intercepted and neutralized by the mobile protection code according to set protection policies. *See id.* at 3:22–40.

E. The ’154 Patent

The ’154 patent is titled “System and Method for Inspecting Dynamically Generated Executable Code” and was issued on March 20, 2012. Ex. 5 to Hannah Decl. (the ’154 patent), ECF 100-7. The patent concerns “new behavioral analysis technology [that] affords protection against dynamically generated malicious code,” which are viruses generated at runtime. *Id.* at 4:32–34; *see also id.* at 3:32–33. In some embodiments, a gateway computer receives content from the internet, where the content includes a call to an original function and an input. *Id.* at 5:26–32. The gateway computer modifies the received content by replacing the call to the original function with a corresponding call to a substitute function. *Id.* at 5:32–35. The substitute function sends the input to a security computer, which determines whether it is safe to invoke the original function with the input at a client computer. *Id.* at 5:35–43. In this approach, the patent provides technology that protects computers from dynamically generated malicious code.

II. LEGAL STANDARD

A. General Principles

Claim construction is a matter of law. *Markman v. Westview Instruments, Inc.*, 517 U.S. 370, 387 (1996). “It is a ‘bedrock principle’ of patent law that ‘the claims of a patent define the invention to which the patentee is entitled the right to exclude,’” *Phillips v. AWH Corp.*, 415 F.3d 1303, 1312 (Fed. Cir. 2005) (en banc) (internal citation omitted), and, as such, “[t]he appropriate starting point . . . is always with the language of the asserted claim itself,” *Comark Commc’ns, Inc. v. Harris Corp.*, 156 F.3d 1182, 1186 (Fed. Cir. 1998).

Claim terms “are generally given their ordinary and customary meaning,” defined as “the meaning . . . the term would have to a person of ordinary skill in the art in question . . . as of the

effective filing date of the patent application.” *Phillips*, 415 F.3d at 1313 (internal citation omitted). The court reads claims in light of the specification, which is “the single best guide to the meaning of a disputed term.” *Id.* at 1315; *see also Lighting Ballast Control LLC v. Philips Elecs. N. Am. Corp.*, 744 F.3d 1272, 1284-85 (Fed. Cir. 2014) (en banc). Furthermore, “the interpretation to be given a term can only be determined and confirmed with a full understanding of what the inventors actually invented and intended to envelop with the claim.” *Phillips*, 415 F.3d at 1316 (quoting *Renishaw PLC v. Marposs Societa’ per Azioni*, 158 F.3d 1243, 1250 (Fed. Cir. 1998)). The words of the claims must therefore be understood as the inventor used them, as such understanding is revealed by the patent and prosecution history. *Id.* The claim language, written description, and patent prosecution history thus form the intrinsic record that is most significant when determining the proper meaning of a disputed claim limitation. *Id.* at 1315–17; *see also Vitronics Corp. v. Conceptronic, Inc.*, 90 F.3d 1576, 1582 (Fed. Cir. 1996).

Evidence external to the patent is less significant than the intrinsic record, but the court may also consider such extrinsic evidence as expert and inventor testimony, dictionaries, and learned treatises “if the court deems it helpful in determining ‘the true meaning of language used in the patent claims.’” *Phillips*, 415 F.3d at 1318 (quoting *Markman*, 52 F.3d at 980). However, extrinsic evidence may not be used to contradict or change the meaning of claims “in derogation of the ‘indisputable public records consisting of the claims, the specification and the prosecution history,’ thereby undermining the public notice function of patents.” *Id.* at 1319 (quoting *Southwall Techs., Inc. v. Cardinal IG Co.*, 54 F.3d 1570, 1578 (Fed. Cir. 1995)).

B. Means-Plus-Function Claims

Paragraph 6 of 35 U.S.C § 112 provides for means-plus-function claiming: “An element in a claim for a combination may be expressed as a means . . . for performing a specified function . . . and such claim shall be construed to cover the corresponding structure, material, or acts described in the specification and equivalents thereof.”¹ When a claim uses the term “means” to describe a

¹ Paragraph 6 of 35 U.S.C. § 112 was replaced with newly designated § 112(f) when the America Invents Act (“AIA”), Pub. L. No. 112-29, took effect on September 16, 2012. Because the Asserted Patents were filed before that date, the Court refers to the pre-AIA version of § 112.

limitation, it creates a presumption that the inventor used the term to invoke § 112 ¶ 6. *Biomedino v. Waters Technologies*, 490 F.3d 946, 950 (Fed. Cir. 2007). The “presumption can be rebutted when the claim, in addition to the functional language, recites structure sufficient to perform the claimed function in its entirety.” *Id.*

If a court concludes that a claim limitation is a means-plus-function limitation, “two steps of claim construction remain: 1) the court must first identify the function of the limitation; and 2) the court must then look to the specification and identify the corresponding structure for that function.” *Id.* The claim limitation will then be construed to cover that corresponding structure and equivalents thereof. 35 U.S.C § 112 ¶ 6.

III. AGREED CONSTRUCTIONS

The parties agree on the construction of several terms. *See* Joint Claim Construction Statement, ECF 85. The Court accordingly approves and adopts the following constructions:

Term	Agreed Construction
Downloadable (’844 patent, claims 1, 15, 43; ’494 patent, claim 10; ’780 patent, claims 1, 9, 17, 18)	an executable application program, which is downloaded from a source computer and run on the destination computer
Database (’494 patent, claim 10)	a collection of interrelated data organized according to a database schema to serve one or more applications

IV. DISPUTED TERMS IN THE ’844 PATENT

The parties dispute four terms in the ’844 patent. All four terms appear in independent claim 43 which recites:

43. An inspector system comprising:

means for receiving a Downloadable;

means for generating a first Downloadable security profile that identifies suspicious code in the received Downloadable; and

means for linking the first Downloadable security profile to the Downloadable before a web server makes the Downloadable available to web clients.

’844 patent at 14:34–42 (emphasis added). The fourth term, also emphasized above in claim 43, is

1 “before a web server makes the Downloadable available to web clients.” That disputed term is
2 recited in claims 1, 15, and 43.

3 Cisco’s briefing raises a threshold issue regarding the first to third disputed terms recited in
4 claim 43. Responsive Br. 1–3, ECF 112. Specifically, Cisco contends that claim 43 is directed to
5 the “inspector 125” embodiment as opposed to the “network gateway 110” embodiment in
6 the ’844 patent. *Id.* Finjan disagrees. Reply Br. 1–3, ECF 127. Because the parties’
7 disagreement pertains to a common issue in the disputed terms, the Court addresses that issue first.

8 Cisco argues that claim 43 is an “inspector system claim and cannot be read onto a
9 network gateway” for several reasons. Cisco first contends that the preamble of claim 43 recites
10 “an *inspector* system” and that Finjan added this word during prosecution. Responsive Br. 2
11 (emphasis in original). According to Cisco, Finjan argued before the patent examiner that “the
12 claims were distinct from [the] Ji [prior art reference] because Ji disclosed a network gateway
13 whereas the claims are directed to an ‘inspector.’” *Id.* (citing Ex. 1 to Gunther Decl. (5/3/2000
14 Response), ECF 112-3). On this basis, Cisco claims that Finjan manifested a clear intention to
15 limit the claim scope to “inspector 125” while excluding gateway embodiments. *Id.* at 2–3.

16 Cisco then asserts that claim 43 requires the inspector system to generate a Downloadable
17 Security Profile (“DSP”) and link the DSP to the Downloadable before a web server makes the
18 Downloadable available to web clients. Responsive Br. 3. In Cisco’s view, only the inspector 125
19 is described in the specification as both “generating” and “linking” the DSP. *Id.* Cisco further
20 asserts that the generic protection engine of the network gateway only generates the DSP and
21 passes the Downloadable without linking the Downloadable to the DSP. *Id.* According to Cisco,
22 because claim 43 recites functions that are performed only by inspector 125, the corresponding
23 structure of the first to third terms must be a structure within the inspector 125. *Id.*

24 Finjan responds that Cisco’s construction is inconsistent with decisions issued by courts
25 within this District. *See* Reply Br. 1–3. Finjan also argues that the ’844 patent discloses
26 embodiments “where the inspector is at different locations, including at the network gateway.” *Id.*
27 at 1.

28 After carefully reviewing the parties’ briefing and the record, the Court agrees with

Finjan’s position. First, Cisco’s reliance on the fact that the preamble of claim 43 recites “inspector” is unpersuasive. “[W]hen a patentee defines a structurally complete invention in the claim body and uses the preamble only to state a purpose or intended use for the invention, the preamble is not a claim limitation.” *Novatek, Inc. v. Sollami Co.*, 559 F. App’x 1011, 1015 (Fed. Cir. 2014) (internal quotation marks and citation omitted). That said, “clear reliance on the preamble during prosecution to distinguish the claimed invention from prior art transforms the preamble into a claim limitation.” *Catalina Mktg. Int’l, Inc. v. Coolsavings.com, Inc.*, 289 F.3d 801, 808 (Fed. Cir. 2002). Here, the Court finds that the preamble of claim 43 is not a limitation. The body of the claim structurally defines the claimed invention. In addition, the prosecution history of the ’844 patent does not clearly show that Finjan distinguished the Ji reference by adding “inspector” to the preamble. Rather, Finjan emphasized that the Ji reference is distinguishable because it “does not teach generating [DSP] or linking the [DSP] to a Downloadable before the web server makes the Downloadable security profile available to web clients” and fails to disclose “examining an already linked [DSP] by network gateways.” See Ex. 1 to Gunther Decl. (5/3/2000 Response) (emphasis in original). These statements do not pertain to the preamble. Thus, the prosecution history does not show that Finjan clearly relied on the preamble to limit claim 43. *Catalina*, 289 F.3d at 808.

To the extent that Cisco invokes the prosecution disclaimer doctrine, its argument fails. Any disclaimer must be “clear and unmistakable” and cannot be “amenable to multiple reasonable interpretations.” *Omega Eng’g, Inc. v. Raytek Corp.*, 334 F.3d 1314, 1325–26 (Fed. Cir. 2003). Here, at best, Finjan’s patent prosecution statements are ambiguous and thus do not support Cisco’s position. As such, Finjan’s statements do not show that it “clearly and unmistakabl[y]” disavowed “network gateway” embodiments from the claim scope. *Id.* In fact, other courts have reached the same conclusion after reviewing the prosecution history of the ’844 patent. See *Finjan, Inc. v. Symantec Corp.*, No. 14-CV-02998-HSG, 2017 WL 550453, at *16 (N.D. Cal. Feb. 10, 2017) (“The Court does not read this language to establish . . . that an inspector by definition can never be at the gateway, or to amount to a clear and unmistakable disavowal.”).

Second, the Court is unpersuaded by Cisco’s argument that claim 43 excludes “network

gateway” embodiments on the grounds that the specification discloses only inspector 125 to “generate” and “link” the DSP. To be sure, the ’844 patent expressly discloses that content inspection engine 160 in the inspector 125 of Fig. 1 to both generate and link DSP to a Downloadable. *See e.g.*, ’844 patent at 8:36–9:4. However, the specification also explicitly describes that the content inspection engine 525 which may be located at the network gateway generates DSP for a Downloadable and that the content inspection engine 525 is “similar to the content inspection engine 160 of the inspector 125.” *Id.* at 7:62–64, 7:67–8:2. The specification further explains that the content inspection engine 525 generates DSP for the received Downloadable as described with reference to Figs. 4 and 6, *id.* at 9:63–65, where it is disclosed that content inspection engine 160 attaches DSP, *id.* at 7:13–16, 8:36–9:4. Based on those disclosures, and in particular, given that the ’844 patent expressly describes that the content inspection engine 160 at the network gateway is similar to those in the inspector 125, a person of ordinary skill in the art would understand that the content inspection engine located at the gateway can link DSP to the Downloadable. The Court therefore rejects Cisco’s argument that the “linking” limitation in claim 43 shows that “network gateway” embodiments are excluded.

Accordingly, the Court concludes that claim 43 covers embodiments where the inspector is located at the network gateway. This interpretation of claim 43 is consistent with the construction of other courts. *Symantec*, 2017 WL 550453, at *16.

The Court now addresses the four disputed terms in the ’844 patent separately below.

A. “means for receiving a Downloadable” (claim 43)

Finjan’s Proposal	Cisco’s Proposal	Court’s Construction
<u>Function</u> : receiving a Downloadable	<u>Function</u> : receiving a Downloadable	<u>Function</u> : receiving a Downloadable
<u>Structure</u> : downloadable file interceptor	<u>Structure</u> : inspector 125 that is external to a network gateway	<u>Structure</u> : downloadable file interceptor

There is no dispute that this term is a means-plus-function limitation and that the function is “receiving a Downloadable.” Opening Br. 3, ECF 100; Responsive Br. 3. The Court therefore adopts the function agreed by the parties.

The parties, however, dispute the structure for this limitation. Finjan argues that the proper structure is “downloadable file interceptor” because the ’844 patent disclose this element to perform “the function of receiving a Downloadable.” Opening Br. 3. Finjan points out that Judge Gilliam and Judge Orrick in this District determined that the proper structure is “downloadable file interceptor.” *Id.* (citing *Finjan, Inc. v. Symantec Corp.*, No. 14-CV-02998-HSG, 2017 WL 550453, at *3–4 (N.D. Cal. Feb. 10, 2017); *Finjan, Inc. v. Sophos, Inc.*, No. 14-CV-01197-WHO, 2015 WL 890621, at *8 (N.D. Cal. Mar. 2, 2015) (“[T]he structure for ‘means for receiving a Downloadable’ is unambiguous: the Downloadable file interceptor.”)).

Cisco counters that the structure is the “inspector 125 that is external to a network gateway.” Responsive Br. 3. Cisco’s proposal is based on its argument that claim 43 excludes “network gateway” embodiments. *See id.* at 4. However, as discussed earlier, the Court rejects Cisco’s position that claim 43 does not cover “network gateway” embodiments. Thus, the Court finds that Cisco’s proposed structure is improper.

The remaining issue is whether the “downloadable file interceptor” is the proper structure. The specification clearly discloses that the downloadable file interceptor performs the function of receiving a Downloadable. *See, e.g.*, ’844 patent at 9:22–23. The parties, however, dispute the location of the downloadable file interceptor. Cisco contends that this structure is not part of the inspector 125 on the grounds that it exists only on the network gateway. Responsive Br. 3; Hearing Tr. 35:22–36:3, ECF 132. Thus, according to Cisco, Finjan’s proposed structure would exclude embodiments where the inspector is external to the network gateway. *See* Hearing Tr. 36:1–3. Finjan countered that its construction “would not read out embodiments from the specification” because the “downloadable file interceptor can be at the inspector as well.” *Id.* at 55:13–17, 55:19–56:4. For support, Finjan pointed to col. 7, line 19 to col. 8, line 2 of the specification, which in part states that the “content inspection engine 525 is similar to the content inspection engine 160 of the inspector 125.” Hearing Tr. 55:19–56:2, 57:2–57:19. On this basis, Finjan represented that its proposed construction does not limit claim 43 to only “network gateway” embodiments. *See id.* at 57:20–58:3.

After reviewing the ’844 patent, the Court agrees with Finjan’s conclusion. The

specification describes a generic protection engine that includes the downloadable file interceptor for receiving incoming Downloadables. ’844 patent at 7:41–48. This generic protection engine includes content inspection engine 525 that is similar to the content inspection engine 160 of the inspector 125, which is external to the gateway as depicted in Fig. 1. *Id.* at 7:19–8:2. The specification further describes that the content inspection engine 160 receives Downloadables like the generic protection engine. *See id.* at 4:52–56 (“[T]he Downloadable ID for the Downloadable will be the same each time the content inspection engine 160 (or a protection engine as illustrated in FIG. 5) receives the same Downloadable”). Based on those disclosures, a person of ordinary skill in the art would understand that inspector 125 with the content inspection engine 160 can include the structure of the downloadable file interceptor described in relation to the protection engine in order to receive Downloadables. Accordingly, although the ’844 patent expressly discusses the downloadable file interceptor in connection to embodiments pertaining to the network gateway or computer client (’844 patent at 7:41–44, 9:19–10:23), the Court finds that a person of ordinary skill in the art would understand that the downloadable file interceptor may be located on inspector 125.

The Court also notes that Judge Gilliam and Judge Orrick’s claim construction orders determined the proper structure as the “downloadable file interceptor.” *Symantec Corp.*, , 2017 WL 550453, at *3–4; *Sophos*, 2015 WL 890621, at *8. The Court’s above conclusion is consistent with those orders. Thus, the Court is further persuaded that the “downloadable file interceptor” is the correct structure for the “means for receiving a Downloadable.” *Symantec*, 2017 WL 550453, at *3 (explaining that prior claim construction orders issued in the same jurisdiction may receive deference). The Court finds no reason to depart from the prior claim construction orders.

For the above reasons, the Court adopts Finjan’s construction.

//

//

//

B. “means for generating a first Downloadable security profile that identifies suspicious code in the received Downloadable” (Claim 43)

Finjan’s Proposal	Cisco’s Proposal	Court’s Construction
<u>Function</u> : generating a first Downloadable security profile that identifies suspicious code in the received Downloadable	<u>Function</u> : generating a first [Downloadable security profile that identifies suspicious code in the received Downloadable] ²	<u>Function</u> : generating a first Downloadable security profile that identifies suspicious code in the received Downloadable
<u>Structure</u> : content inspection engine programmed to perform the algorithm disclosed at Col. 8, lines 51–60 of the ’844 patent	<u>Structure</u> : content inspection engine 160 of inspector 125 programmed to perform the algorithm disclosed at 8:49–60 of the ’844 patent and 5:36–47, 5:54–57, 8:23–24, 9:20–42, and Fig. 7 of U.S. Patent No. 6,092,194	<u>Structure</u> : content inspection engine programmed to perform the algorithm disclosed at Col. 8, lines 51–60 of the ’844 patent

The parties do not dispute that this term is a means-plus-function limitation and that the function is “generating a first Downloadable security profile that identifies suspicious code in the received Downloadable.” Opening Br. 5; Responsive Br. 4. The Court therefore adopts the function agreed by the parties.

The parties, however, diverge on the proper structure for this limitation. Finjan asserts that its proposed structure is correct because the structure is “taken directly from the specification of the ’844 patent as it relates to generating a [DSP].” Opening Br. 5. Finjan further asserts that Judge Gilliam adopted the same structure in *Symantec*, 2017 WL 550453, at *6–7. Opening Br. 5–6.

Cisco first responds that the proper structure should be “tied to inspector 125” but that Finjan’s proposal fails to do so. Responsive Br. 4. Cisco next argues that the specification of the ’844 patent “provides almost no detail of how the content inspection engine 160 generates a DSP that *identifies suspicious code* in the received Downloadable” and that code is distinct from operations. *Id.*; *see also id.* n.4. Based on this assertion, Cisco contends that the Court must look to U.S. Patent Application No. 08/964,388 (now U.S. Patent No. 6,092, 194 (“the ’194 patent”)),

² Cisco states that the square brackets shown in its proposed constructions indicate that the bracketed phrase should be subject to the proposed construction for the respective term in the Joint Claim Construction Statement. Response Br. 4 n.2.

which the '844 patent incorporates by reference, to construe the proper structure. *Id.* at 4–5.

Cisco's first contention is based on the argument that claim 43 excludes "network gateway" embodiments. But, again, the Court rejects Cisco's argument that claim 43 does not cover "network gateway" embodiments. The Court therefore is unpersuaded by Cisco's first contention that the proper structure should be tied to inspector 125.

Regarding Cisco's second contention, the Court is unconvinced by Cisco's reliance on the '194 patent even if that patent were properly incorporated by reference into the '844 patent. The '194 patent does not mention "content inspection engine" and the Court is unable to identify with particularity which algorithms disclosed in the '194 patent pertain to the "content inspection engine" described in the '844 patent. On the other hand, the '844 patent itself sufficiently discloses the algorithm for the content inspection engine that performs the agreed upon function:

As stated above, generating a DSP [by the content inspection engine] includes examining the Downloadable 205 (and the Downloadable components) for all suspicious operations that will or may be performed by the Downloadable, all suspicious code patterns, all known viruses, etc. Generating a DSP may include comparing all operations that will or may be performed against a list of suspicious operations or against a list of rules, e.g., a rules base 165. Accordingly, if an operation in the Downloadable 205 matches one of the suspicious operations or violates one of the rules, then the operation is listed in the DSP 215.

'844 patent at 8:51–60. Similar narrative algorithms have been found to disclose sufficient structure. *See Typhoon Touch Techs., Inc. v. Dell, Inc.*, 659 F.3d 1376, 1385–86 (Fed. Cir. 2011). As such, the Court finds that the proper algorithm is disclosed in col. 8, lines 51–60 of the '844 patent. Indeed, Judge Gilliam reached the same conclusion. *Symantec*, 2017 WL 550453, at *6–7 (declining to rely on the '194 patent and holding that the '844 patent disclosed sufficient structure).

Moreover, Cisco's argument that Court must look to the '194 patent on the grounds that code is distinct from operations is unavailing. Finjan's proposed algorithm in the '844 patent describes "generating a DSP includes examining the Downloadable . . . for all suspicious operations that will or may be performed by the Downloadable, *all suspicious code patterns*, all known viruses, etc." '844 patent at 8:51–55 (emphasis added). As an example, the '844 patent discloses that a DSP can be generated using a "rules base." *Id.* at 8:55–58. Moreover, the

specification discloses that a set of rules may include a “list of suspicious code patterns.” *Id.* at 2:7–8. Thus, although Cisco is correct that “code” is distinct from “operations,” its argument is unavailing. Based on those disclosures, the Court finds that a person of ordinary skill in art would understand how the content inspection engine performs the function based on the algorithm disclosed in col. 8, lines 51–60 of the ’844 patent.

Accordingly, the Court adopts Finjan’s proposed construction.

C. “means for linking the first Downloadable security profile to the Downloadable before a web server makes the Downloadable available to web clients” (Claim 43)

Finjan’s Proposal	Cisco’s Proposal	Court’s Construction
<p><u>Function</u>: linking the first Downloadable security profile to the Downloadable before a web server makes the Downloadable available to web clients</p> <p><u>Structure</u>: content inspection engine programmed to perform the algorithm of step 630 disclosed at Fig. 6; col. 8, lines 65–67; and col. 6, lines 13–24.</p>	<p><u>Function</u>: linking the first Downloadable security profile to the Downloadable [before a web server makes the Downloadable available to web clients]</p> <p><u>Structure</u>: content inspection engine 160 of inspector 125 programmed to perform step 630 of FIG. 6, disclosed at 8:65–67 and col. 6, lines 13–20. Specifically, attaching a Downloadable security profile to the Downloadable (col. 8, lines 65–67 and col. 6, lines 13–18) or attaching to the Downloadable a pointer that points to a stored Downloadable security profile (col. 6, lines 18–20).</p>	<p><u>Function</u>: linking the first Downloadable security profile to the Downloadable before a web server makes the Downloadable available to web clients</p> <p><u>Structure</u>: content inspection engine programmed to perform step 630 of FIG. 6, disclosed at 8:65–67 and col. 6, lines 13–20. Specifically, attaching a Downloadable security profile to the Downloadable (col. 8, lines 65–67 and col. 6, lines 13–18) or attaching to the Downloadable a pointer that points to a stored Downloadable security profile (col. 6, lines 18–20).</p>

There is no dispute that this term is a means-plus-function limitation and that the function is “linking the first Downloadable security profile to the Downloadable before a web server makes the Downloadable available to web clients.” Opening Br. 7; Responsive Br. 5. The Court therefore adopts the agreed upon function.

The parties disagree on the structure for this limitation. Finjan argues that its proposed construction is the one adopted by Judge Gilliam in *Symantec*, 2017 WL 550453, at *8. Opening

Br. 7. Finjan further contends that its proposal is consistent with the scope of the agreed function and does not incorporate unnecessary and confusing elements. *Id.* at 8.

Cisco counters that its proposal specifies that the content inspection engine must be part of the inspector 125. Responsive Br. 6. However, like Cisco’s earlier arguments, this contention is based on Cisco’s view that claim 43 excludes “network gateway” embodiments. The Court rejects that view and thus is unconvinced by this counter-argument. Thus, to the extent that its proposal references content inspection engine 160 and inspector 125, Cisco’s construction is improper.

Cisco also argues that the specification discloses only two methods for linking the DSP to the Downloadable: “(1) attaching the DSP to the Downloadable and (2) attaching a pointer to the Downloadable that points to the DSP stored in the network system 100.” Responsive Br. 6. Cisco further asserts that col. 6, lines 20–24 of the specification should be excluded from Finjan’s proposal. *See id.* That portion of the ’844 patent states the following:

The term “linking” herein will be used to indicate an association between the Downloadable 205 and the DSP 215 (including using a pointer from the Downloadable 195 to the DSP 215, attaching the DSP 215 to the Downloadable 205, etc.)

’844 patent at 6:20–24. Cisco argues that this portion of the specification only identifies the two methods for linking (which are already disclosed in col. 6, lines 13–20 and col. 8, lines 65–67) and that the reference to “an association” and “etc.” are not algorithms that provide corresponding structure. *See* Responsive Br. 6. As discussed below, the Court agrees with Cisco on this point.

In allowing means-plus-function limitations, “Congress . . . plac[ed] specific constraints on how such a limitation is to be construed, namely, by restricting the scope of coverage to *only the structure, materials, or acts* described in the specification as corresponding to the claimed function and equivalents thereof.” *Williamson v. Citrix Online, LLC*, 792 F.3d 1339, 1347 (Fed. Cir. 2015). Here, the parties both propose algorithms for the structure and thus agree that the specification must describe an algorithm to sufficiently disclose the structure. *See Aristocrat Techs. Australia Pty Ltd. v. Int’l Game Tech.*, 521 F.3d 1328, 1333 (Fed. Cir. 2008) (“[C]omputer-implemented means-plus-function term is limited to the corresponding structure disclosed in the specification and equivalents thereof, and the *corresponding structure is*

1 *the algorithm.*” (citation omitted) (emphasis added)). In other words, the structure of the disputed
2 means-plus-function term at issue is limited to the algorithm disclosed in the specification. *See id.*

3 Here, as Cisco argues, col. 6, lines 20–24 of the specification merely restates the function
4 “linking the first Downloadable security profile to the Downloadable” by referring “linking” as an
5 “association” but fails to specify *how* the content inspection engine performs the claimed function.
6 Thus, that passage does not adequately disclose an algorithm (other than the two methods which
7 Cisco does not dispute). *See Aristocrat*, 521 F.3d at 1334 (specification did not contain an
8 algorithm sufficient to disclose structure, when language identified by plaintiff “simply describe[d]
9 the function to be performed, not the algorithm by which it is performed”). In fact, Finjan’s expert,
10 Dr. Medvidovic, testified that the ’844 patent uses “linking” and creating an “association” as
11 synonyms. Ex. 3 to Gunther Decl. (Medvidovic Dep. Tr. 28:17–29:2), ECF 112-5. Because col. 6,
12 lines 20–24 does not disclose anything beyond the two methods of linking by attaching the DSP or
13 the pointer to the DSP, the Court finds that this portion of the specification does not disclose
14 additional algorithm that corresponds to the structure of the means-plus-function claim.

15 Finjan’s counter-arguments are unpersuasive. First, Finjan asserts that Judge Gilliam
16 adopted its proposed construction including col. 6, lines 20–24 of the ’844 patent. However,
17 Judge Gilliam expressed that the sentence “[t]he term ‘linking’ herein will be used to indicate an
18 association between the Downloadable 205 and the DSP 215 (including using a pointer from the
19 Downloadable 195 to the DSP 215, attaching the DSP 215 to the Downloadable 205, etc.)”
20 appears to “simply restate the function ‘linking the first Downloadable security profile to the
21 Downloadable’ rather than providing an algorithm for *how* that function is accomplished, an
22 approach the Federal Circuit has found to be insufficient.” *Symantec*, 2017 WL 550453, at *8.
23 Nevertheless, Judge Gilliam adopted that passage as part of the construction because the parties’
24 proposal covered that portion of the ’844 patent. *Id.* The Court need not give much deference to
25 Judge Gilliam’s construction when he identified an issue with the parties’ proposal but
26 nevertheless accepted it.

27 Second, Finjan argues that Cisco’s own expert, Dr. Clark, admits that “the ’844 patent does
28 not require ‘attaching’ . . . [and] that the function can also be done by linking.” Reply Br. 5.

(citing Ex. 1 to Hannah Decl. (Clark Dep. Tr. 83:3–85:9, 151:17–20), ECF 127-1)). Dr. Clark’s deposition indicates that he explained the two methods for linking based on use of the pointer and attaching the DSP as disclosed in col. 6, lines 13–20. As such, this argument does not change the Court’s conclusion.

Third, Finjan contends that “one skilled in the art would recognize that the DSP 215 can be linked to the Downloadable 205 using other techniques such as association.” Reply Br. 5. However, Finjan’s own expert, Dr. Medvidovic, testified that “association” is synonymous with “linking.” Ex. 3 to Gunther Decl. (Medvidovic Dep. Tr. 28:17–29:2). The Court thus finds that the specification’s mere reference to “association” does not sufficiently disclose an algorithm to support the disputed means-plus-function claim.

Fourth, Finjan argues that it would be improper to limit the term to examples of linking by “attaching the DSP and the pointer.” Reply Br. 5. However, the Court’s rejection of Finjan’s position is not improper. As mentioned above, section 112 ¶ 6 requires that a means-plus-function claim to cover “only the structure, materials, or acts described in the specification.” *Williamson*, 792 F.3d at 1347. The ’844 patent discloses only the two methods identified in col. 6, lines 13–20 and col. 8, lines 65–67 as the algorithm, and thus the scope of the dispute term must be limited to that disclosure. *See id.*

Based on the foregoing, the Court adopts Cisco’s proposed structure with the exception of the reference to inspector 125 as set forth in the following: “content inspection engine programmed to perform step 630 of FIG. 6, disclosed at 8:65–67 and col. 6, lines 13–20. Specifically, attaching a Downloadable security profile to the Downloadable (col. 8, lines 65–67 and col. 6, lines 13–18) or attaching to the Downloadable a pointer that points to a stored Downloadable security profile (col. 6, lines 18–20).”

**D. “before a web server makes the Downloadable available to web clients”
(Claims 1, 15, 43)**

Finjan’s Proposal	Cisco’s Proposal	Court’s Construction
Plain and ordinary meaning. Plain and ordinary meaning of “web client” is “an application on the computer of an end user that requests a Downloadable from the web server”	before any web server makes the Downloadable available to any web client	Plain and ordinary meaning. Plain and ordinary meaning of “web client” is “an application on the computer of an end user that requests a Downloadable from the web server”

Finjan argues that its proposed construction is proper because it is supported by the intrinsic record and is consistent with the main purpose of the '844 patent. Opening Br. 9–10 (citing '844 patent at 1:62–63, 2:65–3:2). Finjan further urges this Court to adopt Finjan's construction as it is exactly the same one adopted by Judge Gilliam in *Symantec* and is consistent with decision of the Federal Circuit and numerous other orders in this District. *Id.* at 9 (citing *Symantec*, 2017 WL 550453, at *18); *id.* (collecting cases). In particular, Finjan asserts that Cisco's construction directly contradicts the Federal Circuit decision in the appeal of *Finjan, Inc. v. Blue Coat Sys., Inc.*, No. 13-cv-03999-BLF ("*Blue Coat I*"), 2014 WL 5361976 (N.D. Cal. Oct. 20, 2014). Opening Br. 11 (citing *Finjan, Inc. v. Blue Coat Sys., Inc.*, 879 F.3d 1299, 1306–07 (Fed. Cir. 2018)). Moreover, Finjan claims that Cisco's proposal improperly excludes "network gateway" embodiments. *See id.* at 12.

Cisco vigorously disagrees. Cisco argues that the claim language requires that a DSP be generated and linked to the Downloadable "'before' a web server takes a certain action, i.e., makes the Downloadable available to web clients." Responsive Br. 8. According to Cisco, a Downloadable is made available to web clients "when the web server places (e.g., *deploys*) the Downloadable on the Internet such that web clients, or even a network gateway, may request the Downloadable." *Id.* (emphasis added). Cisco points to the use of "deployment" in the specification to support its proposed construction. *Id.* (citing '844 patent at 5:3–13). Cisco also asserts that a similar construction was recently adopted in *Finjan, Inc. v. ESET, LLC*, No. 3:17-cv-0183-CAB-(BGS) (S.D. Cal.). *Id.* at 9. As another point, Cisco contends that Finjan's statements that were made during prosecution to distinguish the Ji prior art reference support Cisco's construction. *Id.* (Ex. 1 to Gunther Decl. (5/3/2000 Response) 2–5).

The Court is unpersuaded by Cisco's arguments. First, as Finjan argues, Cisco's interpretation of the disputed term would exclude embodiments where inspection of the Downloadable occurs at a network gateway. Cisco asserts that this is not a problem because "the claims at issue are *inspector claims*—not network gateway claims." Responsive Br. 10 (emphasis in original). However, as discussed earlier, the Court rejects such an interpretation because the record does not clearly show that "network gateway" embodiments are excluded for the claims at

1 issue.

2 Second, Cisco’s argument regarding the “deployment” language in the ’844 patent has
3 been rejected by courts in this District. In particular, Judge Gilliam in *Symantec* held that the
4 “deployment” language did not support a construction similar to what Cisco proposes here. 2017
5 WL 550453, at *14–16. In doing so, Judge Gilliam held that such a construction would
6 improperly exclude preferred embodiments from the claims and that “this is not a circumstance in
7 which it may properly interpret the asserted claims to exclude what [the defendant] characterizes
8 as the ‘gateway embodiment,’ because that embodiment is not ‘inconsistent with unambiguous
9 language in the patent’s specification or prosecution history.’” *Symantec*, 2017 WL 550453, at
10 *16. Moreover, Judge Gilliam found that the ’844 patent’s prosecution history regarding the Ji
11 reference did not establish that “an inspector by definition can never be at the gateway, or to
12 amount to a clear and unmistakable disavowal of claim scope.” *Id.* (citing *SAS Inst., Inc. v.*
13 *ComplementSoft, LLC.*, 825 F.3d 1341, 1349 (Fed. Cir. 2016)). The Court agrees and finds no
14 reason to depart from Judge Gilliam’s conclusions.

15 Third, Cisco’s reliance on *ESET* is weak. In *ESET*, the Southern District of California
16 court adopted the following construction: “before the Downloadable is available on a web server
17 to be called up or forwarded to a web client.” *ESET*, No. 3:17-cv-0183-CAB-(BGS), Dkt. 195.
18 The court, however, provided no reasoning for its construction. Moreover, *ESET*’s claim
19 construction ruling is from outside this District. Under these circumstances, the Court gives little
20 weight to the construction in *ESET* compared to the thoroughly reasoned claim construction orders
21 of this District. *Symantec*, 2017 WL 550453, at *3 (“If anything, to the extent possible, the degree
22 of deference should be greater where the prior claim construction order was issued in the same
23 jurisdiction.”); *Visto Corp. v. Sproqit Techs., Inc.*, 445 F. Supp. 2d 1104, 1107–08 (N.D. Cal.
24 2006) (explaining that the Supreme Court has stressed the particular importance of
25 intrajurisdictional uniformity in claim construction).

26 Fourth, the Federal Circuit’s decision in *Finjan, Inc. v. Blue Coat Sys., Inc.*, 879 F.3d 1299
27 (Fed. Cir. 2018) reinforces the Court’s view that Finjan’s proposal is the proper construction. In
28 that decision, the Federal Circuit was deciding whether there was substantial evidence to support

the jury verdict of infringement based on the claim construction issued by the trial court. *Id.* at 1306–07. The issued construction “before [a/the] non-network gateway web server make[s] the Downloadable available to web clients” (*id.* at 1306) is similar to Finjan’s proposal in this case except for the addition of “non-network gateway,” which was added in *Blue Coat I* due to a dispute not present here. To be clear, the issue of claim construction was not squarely presented before the Federal Circuit because the defendant waived its argument. *Id.* Nevertheless, in reaching its decision, the Federal Circuit held that the disputed term could “reasonably be understood to require that linking occur at some point before users are *permitted to access* that downloadable—but not necessarily before the downloadable is made available on the Internet.” *Id.* (emphasis in original). Thus, the Federal Circuit provided guidance on the reasonable interpretation of the disputed term. As Finjan argues (Opening Br. 11), that guidance directly contradicts the interpretation that Cisco urges this Court to adopt. The Court therefore is further unconvinced by Cisco’s proposed construction.

Accordingly, the Court rejects Cisco’s proposal and adopts the plain and ordinary meaning of the disputed term as other courts within this District have done so. *Symantec*, 2017 WL 550453, at *14–16.

One remaining issue is whether the construction should clarify the plain and ordinary meaning of “web client” in the disputed term. Finjan urges this Court to construe “web client” to mean “an application on the computer of an end user that requests a Downloadable from the web server” as was determined in *Symantec*. Opening Br. 9. Cisco provides no counter-argument as to this issue. In *Symantec*, Judge Gilliam found that the plain and ordinary meaning of “web client” in the context of the ’844 patent is “an application on the computer of an end user that requests a Downloadable from the web server” and that instructing this construction to the jury was necessary to avoid a dispute at trial. 2017 WL 550453, at *16–17. The Court agrees with Judge Gilliam’s reasoning and therefore adopts Finjan’s proposed construction of the phrase “web client.”

//

//

V. DISPUTED TERM IN THE '494 PATENT

The parties dispute one term in the '494 patent, which is addressed below.

A. “Downloadable scanner” (Claim 10)

Finjan’s Proposal	Cisco’s Proposal	Court’s Construction
No construction necessary – Plain and ordinary meaning.	software for disassembling machine code of a Downloadable and resolving commands to identify suspicious operations in the Downloadable	software that searches code to identify suspicious patterns or suspicious computer operations

The disputed term “Downloadable scanner” appears in independent claim 10, which recites:

10. A system for managing Downloadables, comprising:

a receiver for receiving an incoming Downloadable;

a **Downloadable scanner** coupled with said receiver, for deriving security profile data for the Downloadable, including a list of suspicious computer operations that may be attempted by the Downloadable; and

a database manager coupled with said Downloadable scanner, for storing the Downloadable security profile data in a database.

'494 patent at 22:7–16 (emphasis added).

Finjan argues that “Downloadable scanner” should be given its plain and ordinary meaning because “it is a simple term that a person of ordinary skill in the art[] . . . would understand without further construction.” Opening Br. 13. Finjan further asserts that the claim language is clear and unambiguous because claim 10 describes that the Downloadable scanner “derives security profile (“DSP”) data for a Downloadable, including a list of suspicious computer operations that the Downloadable may attempt.” *Id.*

Cisco responds that its proposed construction follows directly from the Court’s decision in *Finjan, Inc. v. Blue Coat Sys., LLC*, No. 15-CV-03295-BLF, 2016 WL 7212322, at *1 (N.D. Cal. Dec. 13, 2016) (“*Blue Coat II* Section 101 Order”). Responsive Br. 11–12. Specifically, Cisco asserts that the Court noted that the '494 patent does not include relevant disclosure concerning a “Downloadable scanner” and thus looked at U.S. Patent No. 6,092,194 (“the '194 patent”) which is incorporated by reference in the '494 patent. *Id.* at 12. According to Cisco, the '194 patent

discloses a code scanner and a decomposition process where the code scanner “disassembl[es] the machine code of the Downloadable” and “resolv[es] a respective command in the machine code.” *Id.* In light of the disclosure in the ’194 patent, Cisco argues that the Court determined that the claims in the ’494 patent were directed to patent-eligible subject matter because they require “a precise process of decomposing code and extracting operations.” *Id.* at 11.

In Finjan’s view, Cisco’s proposed construction is contrary to the intrinsic record and unreasonably limits the claim because the ’494 patent discloses that Downloadables may include “scripting languages such as HTML, JavaScript and program script.” Opening Br. 13–14. Finjan also counters that there is no evidence that Finjan unequivocally disavowed embodiments where a Downloadable is not in machine code. *See* Reply Br. 8. Moreover, Finjan asserts that Cisco’s reliance on the *Blue Coat II* Section 101 Order is misplaced because the “standard for claim construction and a Section 101 analysis are different as a finding that a claim is non-abstract by looking at the specification to determine the problem to be solved does not mean there has been a clear and unmistakable disclaimer in the intrinsic record.” *Id.*

As a preliminary issue, the Court agrees that it must look to the ’194 patent to construe “Downloadable scanner.” As noted in the *Blue Coat II* Section 101 Order, “significant clarifying detail [of the claims] is provided in a specification that belongs not to the ’494 patent itself, but to a parent patent (the ’194 patent), which the ’494 patent identifies and declares to be ‘hereby incorporated by reference.’” 2016 WL 7212322, at *7 (citing ’494 patent at 1:37–38). That said, the precise standard applied in the *Blue Coat II* Section 101 Order is not applicable to the claim construction issue presented to this Court. There the court faced the determination of patent-eligibility of the claims and recognized that it was “not required to undertake a full claim construction process in order to do this.” *Id.* Also, the court was ruling on a motion for judgment on the pleadings and thus “err[ed] on the side of incorporating more—not less—particularities from the specification into its understanding of the claims, while still refraining from importing limitations from the specification” in order to construe the pleadings in favor of Finjan. *Id.*, at *8.

After reviewing the specifications of the ’494 and ’194 patents, the Court rejects Cisco’s proposed construction. Cisco’s construction relies on a specific method disclosed in relation to

Fig. 7 of the '194 patent. *See* '194 patent at 9:20–33. To be sure, that method is the only embodiment that discloses a process of decomposing code and extracting operations in detail. However, it is “well established[] [that] an applicant is not required to describe in the specification every conceivable and possible future embodiment of his invention.” *Innogenetics, N.V. v. Abbott Labs.*, 512 F.3d 1363, 1370 (Fed. Cir. 2008) (citing *SRI Int'l v. Matsushita Elec. Corp. of Am.*, 775 F.2d 1107, 1121 (Fed. Cir. 1985)). Indeed, claims of a patent generally should not be construed as being limited to one embodiment even if the patent describes only that embodiment. *See Phillips*, 415 F.3d at 1323. Therefore, the Court is unpersuaded by Cisco’s argument that a “Downloadable scanner” is limited to the embodiment where the code scanner is described to disassemble machine code.

In fact, the specification of the '194 patent shows that Cisco’s construction is too narrow. For instance, the specification discloses an embodiment where a code scanner “uses conventional parsing techniques to decompose the code (including all prefetched components) of the Downloadable.” '194 patent at 5:42–44. Here, there is no limitation that the code must be “machine code.” The specification further discloses that the code scanner “may search the code for any pattern, which is undesirable or suggests that the code was written by a hacker.” *Id.* at 5:54–57. In addition, the specification describes that the Downloadable may include a JavaScript or a Visual Basic script, neither of which is machine code. In light of those disclosures, a person of ordinary skill in the art would understand that the '194 patent describes that the code scanner may evaluate non-machine code in the Downloadable without disassembling machine code.

The above conclusion is not inconsistent with the *Blue Coat II* Section 101 Order. As mentioned, in that case, the court was presented with a different issue and considered the claim based on a standard applicable for a motion for judgment on the pleadings. To be sure, the court recognized that an embodiment of deriving security file data involves “a precise processing of decomposing code and extracting operations” discussed in relation to Fig. 7 of the '194 patent. *Blue Coat II* Section 101 Order 20–21. The court further stated that a person of ordinary skill in the art would understand “deriving security profile data” to “refer to this type of process.” *Id.* at 21. When referencing “this type of process,” the Court recognized that the “move from profiling

at the file level to profiling at the operation level was [a] non-conventional rearrangement of the malware profiling process.” *Id.* Profiling at the operation level does not necessarily require disassembling machine code.³ As such, when the court mentioned “this type of process,” it did not hold that the “*precise* processing of decomposing code and extracting operations” described in connection to Fig. 7 (which discusses disassembly of machine code) was a limitation incorporated into the “Downloadable scanner.”

On the other hand, the Court also finds Finjan’s proposed construction to be unsatisfactory. Presenting that the scanner has a plain and ordinary meaning will not aid the jury. As discussed above, the specification of the ’194 patent discloses that a code scanner “may search the code for any pattern, which is undesirable or suggests that the code was written by a hacker.” *Id.* at 5:54–57. The code scanner may generate DSP data that includes suspicious computer operations. *Id.* at 5:50–54. As such, the Court finds that a person of ordinary skill in the art would understand that a “Downloadable scanner” is “software that searches code to identify suspicious patterns or suspicious computer operations.” The Court therefore will adopt this construction.

During the hearing, Finjan asserted that the ’494 patent “talks about suspicious operations” and a construction would have to take into account for “operations.” Hearing Tr. 85:7–14, 85:22–24. The Court is satisfied that claim 10 with the adopted construction of “Downloadable scanner” provides identification of “suspicious computer operations” as described in relation to the scanner disclosed in the ’494 and ’194 patents.

VI. DISPUTED TERM IN THE ’780 PATENT

The parties dispute one term in the ’780 patent, which is addressed below.

A. “performing a hashing function on the Downloadable and the fetched software components to generate a Downloadable ID” (Claims 1, 9, 17, 18)

Finjan’s Proposal	Cisco’s Proposal	Court’s Construction
performing a hashing function on the Downloadable together with its fetched software	performing a hashing function on the Downloadable together with its fetched software	performing a hashing function on the Downloadable together with its fetched software components to generate a unique and

³ Indeed, during the hearing, Cisco indicated that the machine code is converted to a “different form” to determine whether operations are suspicious. *See* Hearing Tr. 72:1–11.

components to generate a Downloadable ID	components to generate a unique and reproducible number or string that is the same whether the software components are referenced by the Downloadable and or are included with the Downloadable	reproducible ID for that Downloadable
--	---	---------------------------------------

The disputed term appears in independent claims 1, 9, 17 and 18 of the '780 patent.

Claim 1 is representative of how the term is used in the claim language:

1. A computer-based method for generating a Downloadable ID to identify a Downloadable, comprising:

obtaining a Downloadable that includes one or more references to software components required to be executed by the Downloadable;

fetching at least one software component identified by the one or more references; and

performing a hashing function on the Downloadable and the fetched software components to generate a Downloadable ID.

'780 patent at 10:23–32 (emphasis added).

Finjan asserts that its proposed construction should be adopted because other courts accepted the same construction. Opening Br. 14–15. Finjan further contends that Cisco's proposal is not supported by the intrinsic record to the extent that Cisco intends to limit a "Downloadable ID" to be a single "number or string." *Id.* at 15. According to Finjan, the claims allow the generation of one or more Downloadable IDs. *Id.*

Cisco responds that its construction articulates the actual meaning of the claim term. Responsive Br. 13–14. In particular, Cisco asserts that the "alleged invention of the '780 patent is generating a Downloadable ID by performing 'a digital hash of the *complete* Downloadable code' so the Downloadable ID 'will be the same each time the ID generator . . . receives the same Downloadable.'" *Id.* (citing the '780 patent at 4:54–66)) (emphasis in original). Cisco claims that its construction is consistent with the intrinsic record. *Id.* at 14.

The Court is unpersuaded by Cisco's arguments. The specification of the '780 patent clearly discloses an embodiment where an ID generator performs a "hashing function on *at least a portion* of the Downloadable code to generate a Downloadable ID." '780 patent at 9:65–67 (emphasis added). Regarding this embodiment, the specification also discloses that "the

Downloadable ID will be the same for the identical Downloadable each time it is encountered.” *Id.* at 10:3–5. Thus, to the extent that Cisco contends that the complete Downloadable must be always hashed, that view is contrary to the intrinsic record. *Oatey Co. v. IPS Corp.*, 514 F.3d 1271, 1277 (Fed. Cir. 2008) (“At leas[t] where claims can reasonably to interpreted to include a specific embodiment, it is incorrect to construe the claims to exclude that embodiment, absent probative evidence on the contrary.”).

Cisco further argues that Finjan’s proposed construction led to a post-*Markman* dispute on the claim scope in *Finjan, Inc. v. Blue Coat Sys., Inc.*, No. 13-CV-03999-BLF, 2015 WL 3630000, at *5 (N.D. Cal. June 2, 2015). Responsive Br. 13. In that case, the dispute between the parties was the meaning of “together with” in the adopted construction “performing a hashing function on the Downloadable *together with* its fetched software components” *Finjan*, 2015 WL 3630000, at *5. Upon reviewing the specification, the court held that the claim term at issue required a hashing function that “transmutes the Downloadable and its components into a unique and reproducible ID for that Downloadable.” *Id.*, at *7; *see also id.*, at *6. Although Cisco appears to have attempted to address the issue raised in *Blue Coat*, 2015 WL 3630000, the Court finds that Cisco’s construction contains unnecessary language and is confusing. In light of the dispute in *Blue Coat*, 2015 WL 3630000, the Court will adopt the language which both Finjan and Cisco proposes but modify it as the following: “performing a hashing function on the Downloadable together with its fetched software components to generate a unique and reproducible ID for that Downloadable.”

Finjan contends that the claims allow the generation of one or more Downloadable IDs. Opening Br. 15. The Court’s adopted construction does not preclude such a possibility. As discussed in *Finjan*, 2015 WL 3630000, at *7, a sensible reading of the claims “would be that the ID generator performs ‘one or more’ hashing functions to generate ‘one or more’ Downloadable IDs for ‘one or more’ Downloadables.” Insofar as a specific Downloadable is concerned, the specification discloses performing a hashing function so that “the Downloadable ID will be the same [for that] Downloadable.” ’780 patent at 10:3-5. The Court’s construction is consistent with the specification of the ’780 patent.

During the hearing, Cisco asserted that one will not obtain the same Downloadable ID if the Downloadable and its components are individually “hash[ed]” and then combined as opposed to “hash[ing]” the combination of the Downloadable and its components altogether. *See* Hearing Tr. 79:13–16, 80:17–25. However, whether Cisco is correct may depend on the specific hash function that is performed. In any case, Cisco’s argument does not convince this Court that the adopted construction is inconsistent with the record. As discussed above, the specification of the ’780 patent discloses the use of a hashing function that leads to a unique and reproducible ID for a specific Downloadable. *See, e.g.*, ’780 patent at 4:64–66, 10:3-5.

VII. DISPUTED TERMS IN THE ’633 PATENT

The parties dispute three terms in the ’633 patent. All three terms appear in independent claim 13 which recites:

13. A processor-based system for computer security, the system comprising:

means for receiving downloadable-information;

means for determining whether the downloadable-information includes executable code; and

means for causing mobile protection code to be communicated to at least one information-destination of the downloadable-information, if the downloadable-information is determined to include executable code.

’633 patent at 21:49–57 (emphasis added). The Court addresses the three terms separately below.

A. “means for determining whether the downloadable-information includes executable code” (Claim 13)

Finjan’s Proposal	Cisco’s Proposal	Court’s Construction
<u>Function</u> : determining whether the downloadable-information includes executable code	<u>Function</u> : determining whether the downloadable-information includes executable code	<u>Function</u> : determining whether the downloadable-information includes executable code
<u>Structure</u> : code detection engine to perform the algorithm of col. 2 at 63–66, col. 14, line 58 to col. 15, line 8 or col. 16, lines 16–27.	<u>Structure</u> : server with ⁴ content inspection engine for performing algorithm of FIG. 10a described at 19:18–47.	<u>Structure</u> : code detector to perform the algorithm of col. 14, line 58 to col. 15, line 8 or col. 16, lines 16–27.

There is no dispute that this term is a means-plus-function limitation and that the function

⁴ Cisco removed the “server” component in its proposed construction. Responsive Br. 15 n.9.

is “determining whether the downloadable-information includes executable code.” Opening Br. 16; Responsive Br. 15. The Court therefore adopts the agreed upon function.

The parties, however, dispute the structure for this limitation. This disagreement implicates three issues. First, Finjan points to “code detection engine,” whereas Cisco points to “content inspection engine” as the structure. Upon reviewing the ’633 patent, the Court finds that there is no meaningful difference between “code detection engine,” “content inspection engine,” and “code detector.” For instance, the specification describes that Fig. 5 illustrates a “content inspection engine” in the Brief Description of the Drawings Section while describing the element depicted in Fig. 5 as a “code detector” in the Detailed Description Section. ’633 patent at 4:60–61, 12:8. The term “code detection engine” is used in the Summary of the Invention Section once as a general description. *See id.* at 2:64–66. The parties do not provide any argument that there is a meaningful difference between these terms. In fact, Finjan’s brief uses “code detection engine” to describe portions of the specification which are referenced in connection to a “code detector.” During the hearing, Cisco indicated that the three terms appear to indicate “the same thing.” Hearing Tr. 91:18–25. Against this background, the Court adopts “code detector” which the specification explicitly uses to describe the specific algorithms discussed below.

Second, Cisco argues that Finjan’s proposed algorithm at col. 2 at lines 63–66 of the ’633 patent “merely parrots the recited function and does not recite an algorithm.” Responsive Br. 15 n.10. The Court agrees with Cisco’s argument. Column 2 at lines 62–66 of the specification states that the “protection engine includes an information monitor for monitoring information received by the server, and a code detection engine for determining whether the received information includes executable code.” This portion merely repeats the function and does not disclose an algorithm for Finjan’s proposed structure. Thus, the Court rejects Finjan’s proposal to include the “algorithm [at] col. 2 at 63–66.” *See Aristocrat*, 521 F.3d at 1334.

Third, the main dispute between the parties is what kind of algorithm is the “code detector” required to perform.⁵ Finjan proposes that the “code detector” performs either the algorithm at

⁵ Because the Court adopts “code detector,” it will refer to this term throughout this part of the order instead of “code detection engine” or “content inspection engine” proposed by the parties.

col. 14, line 58 to col. 15, line 8 *or* col. 16, lines 16–27 of the specification. Cisco proposes that the “code detector” performs the algorithm of FIG. 10a described at col. 19 lines 18–47 of the specification. During the hearing, Cisco clarified its position and stated that it would accept Finjan’s proposal if the word “or” was changed to “and.” *See* Hearing Tr. 92:23–25, 94:1–3.

Here, the parties agree that the ’633 patent discloses two types of techniques that correspond to the recited function issue: (1) file-type detection (*see* ’633 patent at 14:58–15:8) and (2) content detection (*see id.* at 16:16–27). File-type detection involves evaluating the file format (e.g., exe, applet, zip, etc.) to determine whether the file contains executable code. *Id.* at 14:58–15:8. Content detection concerns, for example, analyzing the file for binary information or patterns to determine whether the file contains executable code. *Id.* at 16:16–27. Here, the parties’ dispute is whether the claim term requires both techniques to be deployed or only one. Cisco contends that both techniques must be utilized.

The specification itself does not disclose that the code detector must perform both techniques. At the hearing, Cisco argued that its proposed construction is based on Finjan’s arguments that were presented during a reexamination proceeding. Hearing Tr. 94:1–3. According to Cisco, Finjan argued during the reexamination proceeding that the claims require “an active determination” which accounts for the possibility that a downloadable has executable code and that content detection is required because “a downloadable might obfuscate that [it] has executable code.” Responsive Br. 16. In Cisco’s view, Finjan disclaimed the use of only deploying file-type detection and therefore claim 10 requires both file-type detection and content detection in the manner described in relation to Fig. 10a. *See id.* at 17.

Prosecution disclaimer precludes a patentee from recapturing a specific meaning that was previously disclaimed during prosecution. *Omega Eng’g, Inc. v. Raytek Corp.*, 334 F.3d 1314, 1325–26 (Fed. Cir. 2003). For prosecution disclaimer to attach, “disavowing actions or statements made during prosecution [must] be both clear and unmistakable.” *Id.* Here, Cisco relies on Finjan’s statements that the claims “require an active determination as to whether (or not) any (not limited to applets) downloadable information contains executable code” and that “[b]y searching for executable code, rather than simply applet tags, the ’633 [p]atent provides additional protection

against obfuscated code because a Downloadable may be malicious even if it does not contain an applet identifier.” Responsive Br. 16 (citing Exs. 6 and 7 to Gunther Decl. (response during reexamination proceeding), ECF 112-8, -9). The Court, however, finds that those statements do not show that Finjan clearly and unmistakably disclaimed file-type detection for the claim scope. For instance, arguably, “active determination” and “searching for executable code, rather than simply applet tags” may include analyzing a file format to determine whether the file includes an executable type such as an “exe” file. In other words, at best, whether Finjan disclaimed embodiments that use only the file-type detection is ambiguous. Accordingly, the Court concludes that Cisco has not established that the prosecution history of the ’633 patent shows disclaimer regarding the file-type detection algorithm. Therefore, the Court rejects Cisco’s proposed construction.

Based on the above discussion, the Court adopts the following structure: “code detector to perform the algorithm of col. 14, line 58 to col. 15, line 8 or col. 16, lines 16–27.”

B. “means for causing mobile protection code to be communicated to at least one information-destination of the downloadable-information, if the downloadable-information is determined to include executable code” (Claim 13)

Finjan’s Proposal	Cisco’s Proposal	Court’s Construction
<u>Function</u> : if the downloadable-information is determined to include executable code, causing mobile protection code to be communicated to at least one information-destination of the downloadable-information without modifying the executable code	<u>Function</u> : causing mobile protection code to be communicated to at least one information-destination of the downloadable-information, if the downloadable-information is determined to include executable code	<u>Function</u> : if the downloadable-information is determined to include executable code, causing mobile protection code to be communicated to at least one information-destination of the downloadable-information without modifying the executable code
<u>Structure</u> : transfer engine programmed to perform the algorithm of col. 14, lines 24–36.	<u>Structure</u> : server with ⁶ packaging engine 403 of FIG. 4 performing the algorithm described at 12:38–12:64 and 13:19–13:41	<u>Structure</u> : transfer engine programmed to perform the algorithm of col. 14, lines 24–36.

⁶ Cisco removed the “server” component in its proposed construction. Responsive Br. 15 n.9.

There is no dispute that this term is a means-plus-function limitation. Regarding the function, Finjan proposes that the language “if the downloadable-information is determined to include executable code” should be placed before “causing mobile protection code to be communicated to at least one information-destination of the downloadable-information without modifying the executable code.”⁷ Opening Br. 18. Cisco contends that the order of the two phrases should be reversed. Responsive Br. 18.

As for structure, Finjan points to “transfer engine” that performs “the algorithm of col. 14, lines 24–36,” whereas Cisco points to “packaging engine 403 of FIG. 4 [that] perform[s] the algorithm described at 12:38–12:64 and 13:19–13:41.” *Compare* Opening Br. 17–18 *with* Responsive Br. 17.

The Court first turns to the structure of the claim term which reflects the parties’ main dispute. Finjan argues that the “transfer engine” is the proper structure because the ’633 patent describes that element to “transfer[] mobile code protection to its information-destination.” Opening Br. 18. Finjan also contends that Cisco’s proposal is incorrect because the portions of the specification identified by Cisco “relate to other components that do not cause mobile protection code to be communicated.” *Id.* As such, in Finjan’s view, Cisco’s proposed structure imports “limitations that are related to other functions and components, such as the agent generator and linking engine.” *Id.* at 19.

Cisco responds that the “causal link between the determination that the downloadable-information includes executable code and the action—causing [mobile protection code] to be communicated—is at the heart of the recited function and must be accounted for in identifying the corresponding structure.” Responsive Br. 18. In this regard, Cisco contends that the transfer engine is not the structure that “causes” communication of the mobile protection code because the transfer engine relies on “results” from the “linking engine,” which is included in the packaging engine. *Id.* at 19. Cisco further asserts that the transfer engine even transfers Downloadables that do not include executable code presumably without the mobile protection code. *Id.* at 18–19.

⁷ Cisco agrees to add the limitation “without modifying the executable code.” Responsive Br. 18 n.15.

After reviewing the '633 patent, the Court finds that a person of ordinary skill in the art would understand that the transfer engine is the proper structure. First, the specification describes that the transfer engine causes protection "results" to be transferred to a destination user device. '633 patent at 14:24–27. Specifically, the specification discloses that the transfer engine can transfer "a Downloadable, a determined non-executable, or a *sandboxed package*." *Id.* at 14:27–29 (emphasis added). A sandboxed package can include mobile protection code and a detected-Downloadable. *Id.* at 11:6–7. Thus, to the extent that Cisco argues that the transfer engine does not "cause" transfer of mobile protection code, that argument is contradicted by the disclosure of the '633 patent.

Second, Cisco improperly relies on the phrase "if the downloadable-information is determined to include executable code." While this phrase states a condition, it does not require that the identified structure take action only when "the downloadable-information is determined to include executable code." As written, the claim language is silent on when the condition is not satisfied. Thus, Cisco's contention that the transfer engine cannot be the structure because that element communicates Downloadables without executable code is irrelevant.

Third, as Finjan argues (Opening Br. 19), Cisco's proposed construction would import extraneous limitations that are limited to other functions performed by elements such as the agent generator and linking engine. While the agent generator provides a mobile protection code and the linking engine forms a sandboxed package ('633 patent at 12:38–64, 13:19–41), those elements do not communicate the mobile protection code to a destination. Thus, adopting Cisco's proposed structure and algorithms would impermissibly import extraneous limitations. *Acromed Corp. v. Sofamor Danek Grp., Inc.*, 253 F.3d 1371, 1382 (Fed. Cir. 2001) ("[A] court may not import into the claim structural limitations from the written description that are unnecessary to perform the claimed function.").

Fourth, other claims in the '633 patent support the Court's conclusion that a person of ordinary skill in the art would understand that the transfer engine is the proper structure that causes mobile protection code to be communicated. For example, independent claim 8 recites similar language to the disputed term at issue:

a **protection agent engine** communicatively coupled to the content inspection engine **for causing mobile protection code (“MPC”) to be communicated** by the computer to at least one information-destination of the downloadable-information, **if the downloadable-information is determined to include executable code.**

’633 patent at 21:27–33. The specification does not disclose that the “protection agent engine” corresponds to the packaging engine, which is Cisco’s proposed structure. On the other hand, the specification clearly describes that the transfer engine is part of the “protection agent engine.” *Id.* at 14:24–27. Thus, when reading claims such as claim 8 and the specification of the ’633 patent, a person of ordinary skill in the art would understand that the transfer engine is the claimed structure that “caus[es] mobile protection code to be communicated to at least one information-destination of the downloadable-information without modifying the executable code” when the condition “if the downloadable-information is determined to include executable code” is satisfied.

Accordingly, the Court accepts Finjan’s proposed construction and adopts “transfer engine programmed to perform the algorithm of col. 14, lines 24–36” as the structure. Regarding function, the parties’ proposed functions are not meaningfully different. The Court accepts Finjan’s proposed language which was previously adopted in *Finjan, Inc. v. Blue Coat Sys., Inc.*, No. 13-CV-03999-BLF, 2014 WL 5361976, at *5 (N.D. Cal. Oct. 20, 2014).

C. “information-destination of the downloadable-information” / “downloadable-information destination” (Claims 1, 8, 13, 14)

Finjan’s Proposal	Cisco’s Proposal	Court’s Construction
No construction necessary – Plain and ordinary meaning. The plain and ordinary meaning of the terms “information-destination” and “downloadable-information destination” is a device or process that is capable of receiving and initiating or otherwise hosting a mobile code execution.	client(s) that originally requested, and is the final destination for, the downloadable-information	user device that includes one or more devices or process that are capable of receiving and initiating or otherwise hosting a mobile code execution

The third disputed term in the ’633 patent is “information-destination of the downloadable-information” and “downloadable-information destination.” The parties’ proposals construe those

two phrases to have the same meaning. *See* Opening Br. 19; Responsive Br. 19. In other words, the parties at least agree that the two phrases should be construed to have the same meaning. *See* Responsive Br. 20 n.17.

Finjan argues that the term should be construed as “a device or process that is capable of receiving and initiating or otherwise hosting a mobile code execution” and that this construction was adopted in *Finjan, Inc., v. Proofpoint, Inc.*, No. 13-CV-05808-HSG, 2015 WL 7770208, at *5 (N.D. Cal. Dec. 3, 2015). Opening Br. 19.

Cisco counters that Finjan’s proposal ignores the concept of the “downloadable-information” recited in the term and that the court in *Proofpoint* analyzed only the meaning of “‘information-destination’ and not ‘information-destination of the downloadable.’” Responsive Br. 19–20 (emphasis in original). In Cisco’s view, the phrase “of the downloadable-information” requires that “the mobile protection code is transmitted to a specific ‘information-destination.’” *Id.* at 20. Cisco argues that the specification makes clear that the final destination for the downloadable-information is the client which originally requested the Downloadable. *Id.* For support, Cisco asserts that the specification describes that Downloadables which are determined *not* to have executable code are delivered to the client that originally requested the downloadable-information. *Id.* at 20–21 (citing ’633 patent at 9:44–57). According to Cisco, “the final destination of a downloadable is the same whether the downloadable is found to contain executable code or not.” *Id.* at 21 (citing ’633 patent at 19:1–17). On this basis, Cisco contends that “the information-destination of the downloadable” is the client that originally requested the downloadable. *Id.* In addition, Cisco argues that Finjan confirmed during prosecution that a downloadable is sent to the final destination along with the mobile protection code. *Id.* at 21–22 (citing Ex. 8 to Gunther Decl. (6/22/2005 Response), ECF 112-10 (“The present invention provides a packaging of mobile protection code *with a downloadable intended for a destination computer* The package is structured so that when it is received at its *intended destination computer*, the mobile protection code is executed prior to executing the downloadable.”)).

Court is unpersuaded by Cisco’s arguments. First, there is no basis for Cisco’s contention that taking into account the language “of the downloadable-information” affects the analysis

1 conducted by the court in *Proofpoint*. In fact, *Proofpoint* addressed the construction of
2 “downloadable-information destination,” which is the very same term at issue here. As such,
3 *Proofpoint*’s construction is persuasive. *Symantec*, 2017 WL 550453, at *3 (“[T]o the extent
4 possible, the degree of deference should be greater where the prior claim construction order was
5 issued in the same jurisdiction.”).

6 Second, as mentioned, Cisco contends that the “information-destination of the
7 downloadable-information” must be the final destination of a downloadable on the grounds that
8 the specification describes that an unexecutable Downloadable is sent to the client that originally
9 requested the downloadable. The Court finds this argument unpersuasive. Even if Cisco’s
10 characterization of the specification is correct (which the Court does not decide), the portions cited
11 by Cisco pertain only to one embodiment disclosed in the ’633 patent. On the other hand, as
12 Finjan asserts, the specification contains other disclosures that support Finjan’s proposal. For
13 example, the specification describes that mobile protection code can be sent to multiple
14 destinations. *See* ’633 patent at 10:11–15. More importantly, the specification explicitly equates
15 “information-destination” with “user device,”⁸ and describes that a “user device” can include “one
16 or more devices or processes (such as email, browser or other clients) that are capable of receiving
17 and initiating or otherwise hosting a mobile code execution.” *Id.* at 7:58–62. In addition, the
18 specification discloses that a “user device” can operate as a firewall/server.” *Id.* at 7:50. The
19 Court thus finds that Cisco’s reading of the specification is too narrow.

20 Third, Cisco’s reliance on the prosecution history of the ’633 patent is unavailing. During
21 prosecution, Finjan described that the “present invention provides a packaging of mobile
22 protection code *with a downloadable intended for a destination computer.*” *See, e.g.,* Ex. 8 to
23 Gunther Decl. (6/22/2005 Response). However, it is ambiguous whether the “intended . . .
24 destination computer” amounts to a “client that originally requested, and is the final destination
25 for, the downloadable-information.” As such, the Court rejects Cisco’s arguments that rely on the
26 prosecution history. *See Inverness Med. Switzerland GmbH v. Warner Lambert Co.*, 309 F.3d

27
28 ⁸ The specification also equates “user devices” with “Downloadable-destinations.” ’633 patent at 7:5.

1373, 1382 (Fed. Cir. 2002) (the ambiguity of the prosecution history made it less relevant to claim construction).

After reviewing the record, the Court finds that the specification’s description of “information-destination” and “user device” to be dispositive:

A suitable information-destination or “user device” can further include one or more devices or processes (such as email, browser or other clients) that are capable of receiving and initiating or otherwise hosting a mobile code execution.

’633 patent 7:58–62. *Vitronics Corp. v. Conceptronic, Inc.*, 90 F.3d 1576, 1582 (Fed. Cir. 1996) (“The specification acts as a dictionary when it expressly defines terms used in the claims or when it defines terms by implication.”). Finjan’s proposal does not reference “user device,” which most accurately reflects the meaning of the disputed term as disclosed in the specification.

Accordingly, the Court adopts Finjan’s proposal with the following modification: “user device that includes one or more devices or process that are capable of receiving and initiating or otherwise hosting a mobile code execution.” This construction is consistent with the holding in *Proofpoint. Symantec*, 2017 WL 550453, at *3 (holding that deference should be given to a prior claim construction order issued in the same jurisdiction).

VIII. DISPUTED TERM IN THE ’154 PATENT

A. “first function” / “second function” (Claim 1)

Finjan’s Proposal	Cisco’s Proposal	Court’s Construction
No construction necessary – Plain and ordinary meaning.	substitute function / original function, which is different than the first function	substitute function / original function, which is different than the first function

The parties dispute the meaning of “first function” and “second “function” in independent claim 1, which recites:

1. A system for protecting a computer from dynamically generated malicious content, comprising:

a content processor (i) for processing content received over a network, the content including a call to a **first function**, and the call including an input, and (ii) for invoking a **second function** with the input, only if a security computer indicates that such invocation is safe;

a transmitter for transmitting the input to the security computer for inspection, when the **first function** is invoked; and

a receiver for receiving an indicator from the security computer whether it is safe to invoke the **second function** with the input.

'154 patent at 17:32–44.

Finjan argues that “first function” and “second function” should be given their plain and ordinary meaning because “a person of ordinary skill in the art understands this term as it is used in the claims and the specification without further construction.” Opening Br. 20. According to Finjan, these phrases have been determined to require no construction in *Finjan, Inc., v. Proofpoint, Inc.*, No. 13-CV-05808-HSG, 2015 WL 7770208, at *8 (N.D. Cal. Dec. 3, 2015). Finjan further asserts that the '154 patent includes embodiments where the “first function and the second function are the same, and not a substitute function.” Opening Br. 20. As an example, Finjan contends that “Document.write(<h>hello</h>)” in Table II is the first function and that “Document.write(<text>)” in Table III is the second function. *Id.* Finjan also points out that the '154 patent describes a recursive function call—Document.write(<h1>Document.write(<h1><SCRIPT>Some JavaScript</SCRIPT></h1>)</h1>”). Reply Br. 15 (citing '154 patent at 12:28–36). According to Finjan, the outer Document.write() function is the first function and the inner Document.write() function is the second function or vice versa.

Cisco counters that Finjan’s proposal renders the terms “first” and “second” superfluous. Responsive Br. 23. Cisco also argues that it would be nonsensical to allow the “first function” and “second function” to be same on the grounds that it would be “impossible to invoke the second function ‘only if’ a security computer indicates such invocation is safe” as the “same” first function would have been already invoked. *Id.* Based on the specification, Cisco claims that the “first function” is the substitute function and that the “second function” is the original function disclosed in the '194 patent. *Id.* at 23–24. In addition, Cisco asserts that Finjan’s statements before the PTAB during an *inter partes review* (“IPR”) proceeding show that the “first function” is the “substitute function.” *Id.* at 24.

As a preliminary issue, the “use of the terms ‘first’ and ‘second’ is a common patent-law convention to distinguish between repeated instances of an element or limitation.” *3M Innovative Properties Co. v. Avery Dennison Corp.*, 350 F.3d 1365, 1371 (Fed. Cir. 2003). As such, the Court is not barred from construing the “first function” and the “second function” as the same type

of function (e.g., Document.write). However, claims are not construed in vacuum. The Court must determine how a person of ordinary skill in the art would understand the claim terms based on the intrinsic and extrinsic record. *Phillips*, 415 F.3d at 1313–18. In doing so, it is entirely appropriate for the Court to “rely heavily on the written description for guidance as to the meaning of the claims” and look to the remainder of the claim language. *Phillips*, 415 F.3d at 1314, 1317; *Linear Tech. Corp. v. Int’l Trade Comm’n*, 566 F.3d 1049, 1059 (Fed. Cir. 2009). Against this backdrop, the Court agrees with Cisco’s proposed construction upon reviewing the record as stated below.

Finjan’s argument that “Table II identifies a first function as Document.write(‘<h>hello<h>’), and Table III identifies a second function as “Document.write(‘text’)” is unsupported by the specification when viewed in light of the language in claim 1. Specifically, claim 1 requires that “a transmitter for transmitting the input to the security computer for inspection, when the *first function* is invoked.” ’154 patent at 17:39–40 (emphasis added). Although Finjan is correct that the specification discusses document.write() in relation to Tables II and III, nowhere does the specification disclose invoking Document.write(‘<h>hello<h>’) (i.e., Finjan’s first function) to transmit the input of the first function to the security computer for inspection. Rather, the specification describes that invoking the *substitute function* (e.g., substitute_document.write()) leads to the transmission of the input to the security computer. ’154 patent at 10:62–64. As such, Finjan’s argument is inconsistent with other limitations of claim 1 and the specification. Moreover, Finjan’s contention that the ’154 patent describes a recursive function call suffers from the same problem. Therefore, the Court finds that a person of ordinary skill in the art would not understand the “first function” and “second function” as Finjan suggests.

Finjan does not point to any disclosure in the ’194 patent that shows that invoking the “first function” leads to “transmi[ssion] [of] the input to the security computer for inspection” where the “first function” and “second function” are the same type of function. To be sure, a court should be careful in confining claims to embodiments disclosed in the specification. *Phillips*, 415 F.3d at 1323. But limitations in the specification may very well define the scope of the claim if that is “how a person of ordinary skill in the art would understand the claim terms.” *See id.*

1 Because the claims and disclosure of the '154 patent supports Cisco's proposed construction and
2 contradicts Finjan's arguments, the Court agrees with Cisco's proposal.

3 The above conclusion is reinforced by Finjan's statements before the PTAB during the IPR
4 proceeding. For example, Finjan argued that "[t]he '154 patent talks about a call to a first
5 function, and that's a call to a substitute function. . . . If you call a substitute function or you call a
6 first function, you don't have the opportunity to call an original function which may harm your
7 computer." Ex. 11 to Gunther Decl. (IPR2016-00151 Tr.) at 35:3-8, ECF 112-13. During the
8 hearing before this Court, Finjan asserted that it was only referring to examples and did not intend
9 to limit "first function" and "second function" to original and substitution functions. Hearing Tr.
10 114:1-9. The transcript of the IPR proceeding, however, reflects that Finjan repeatedly indicated
11 that calling the first function is calling the substitute function (as opposed to calling the original
12 function). Even if Finjan made those representations as examples, the IPR proceeding does not
13 provide any support that the "first function" and "second function" can be the same function such
14 as `document.write()` as Finjan proposes.

15 At the hearing, Finjan further argued that the "'154 patent . . . [does not] require
16 modification of the content" and that "there's nothing about a gateway, any modification
17 happening, and certainly nothing about replacing function calls." Hearing Tr. 104:12-18. It
18 appears that Finjan made this argument to rebut Cisco's construction that the "first function" is the
19 "substitute function." However, the specification clearly discloses a content modifier in the
20 gateway that modifies original content to replace the original function with the substitute function.
21 *See, e.g.,* '154 patent at 9:13-28. In fact, in *Proofpoint*, Finjan made such an argument to address
22 claim construction of the same disputed term. 2015 WL 7770208, at *8 ("Plaintiff notes that . . .
23 the original function is replaced by the substitute function at the gateway, *before* the security
24 computer receives the content." (emphasis in original)). To be clear, claim 1 does not recite a
25 gateway that modifies content. Such an element is not claimed. That said, for the reasons
26 discussed earlier, a person of ordinary skill in the art would understand that the "first function"
27 corresponds to the substitute function in light of the claim language and the specification.

28 For the above reasons, the Court adopts Cisco's proposed construction: the "first function"

means “substitute function,” and the “second function” means “original function, which is different than the first function.” The Court recognizes that this construction is different from the construction provided in *Proofpoint*. The Court respectfully diverges from *Proofpoint* because that case was neither presented with the issue whether the “first function” is the substitute function nor had the benefit of reviewing Finjan’s recent statements made during the IPR proceeding.

IX. ORDER

As set forth above, the Court construes the disputed terms as follows:

Claim Term	Court’s Construction
“means for receiving a Downloadable” (’844 patent, claim 43)	<u>Function</u> : receiving a Downloadable <u>Structure</u> : downloadable file interceptor
“means for generating a first Downloadable security profile that identifies suspicious code in the received Downloadable” (’844 patent, claim 43)	<u>Function</u> : generating a first Downloadable security profile that identifies suspicious code in the received Downloadable <u>Structure</u> : content inspection engine programmed to perform the algorithm disclosed at Col. 8, lines 51–60 of the ’844 patent
“means for linking the first Downloadable security profile to the Downloadable before a web server makes the Downloadable available to web clients” (’844 patent, claim 43)	<u>Function</u> : linking the first Downloadable security profile to the Downloadable before a web server makes the Downloadable available to web clients <u>Structure</u> : content inspection engine programmed to perform step 630 of FIG. 6, disclosed at 8:65–67 and col. 6, lines 13–20. Specifically, attaching a Downloadable security profile to the Downloadable (col. 8, lines 65–67 and col. 6, lines 13–18) or attaching to the Downloadable a pointer that points to a stored Downloadable security profile (col. 6, lines 18–20).
“before a web server makes the Downloadable available to web clients” (’844 patent, claims 1, 15, 43)	Plain and ordinary meaning. Plain and ordinary meaning of “web client” is “an application on the computer of an end user that requests a Downloadable from the web server”
“Downloadable Scanner” (’494 patent, claim 10)	software that searches code to identify suspicious patterns or suspicious computer operations
“performing a hashing function on the Downloadable and the fetched software	performing a hashing function on the Downloadable together with its fetched


components to generate a Downloadable ID” (’780 patent, claims 1, 9, 17, 18)	software components to generate a unique and reproducible ID for that Downloadable
“means for determining whether the downloadable-information includes executable code” (’633 patent, claim 13)	<u>Function</u> : determining whether the downloadable-information includes executable code <u>Structure</u> : code detector to perform the algorithm of col. 14, line 58 to col. 15, line 8 or col. 16, lines 16–27.
“means for causing mobile protection code to be communicated to at least one information-destination of the downloadable-information, if the downloadable-information is determined to include executable code” (’633 patent, claim 13)	<u>Function</u> : if the downloadable-information is determined to include executable code, causing mobile protection code to be communicated to at least one information-destination of the downloadable-information without modifying the executable code <u>Structure</u> : transfer engine programmed to perform the algorithm of col. 14, lines 24–36.
“information-destination of the downloadable-information” / “downloadable-information destination” (’633 patent, claims 1, 8, 13, 14)	user device that includes one or more devices or process that are capable of receiving and initiating or otherwise hosting a mobile code execution
“first function” / “second function” (’154 patent, claim 1)	substitute function / original function, which is different than the first function

The Court also adopts the following constructions that the parties agreed to in their Joint Claim Construction Statement:

Term	Agreed Construction
Downloadable (’844 patent, claims 1, 15, 43; ’494 patent, claim 10; ’780 patent, claims 1, 9, 17, 18)	an executable application program, which is downloaded from a source computer and run on the destination computer
Database (’494 patent, claim 10)	a collection of interrelated data organized according to a database schema to serve one or more applications

IT IS SO ORDERED.

Dated: July 23, 2018


BETH LABSON FREEMAN
United States District Judge